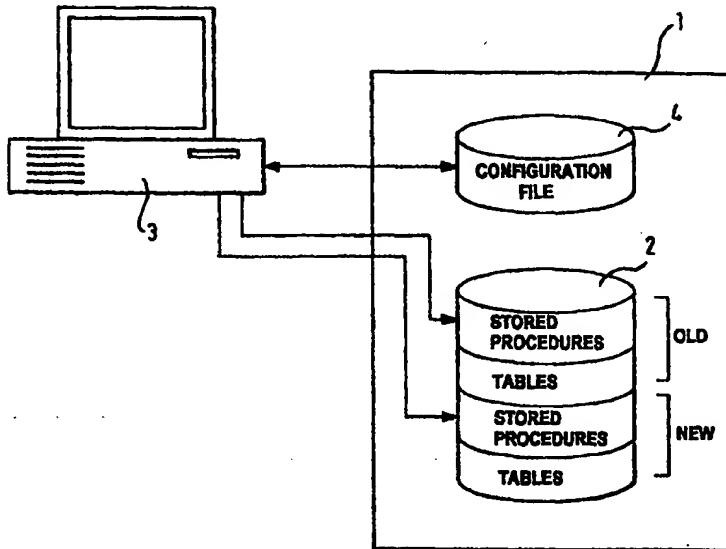




INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : G06F 17/30		A1	(11) International Publication Number: WO 99/08206
			(43) International Publication Date: 18 February 1999 (18.02.99)
<p>(21) International Application Number: PCT/EP98/05026</p> <p>(22) International Filing Date: 7 August 1998 (07.08.98)</p> <p>(30) Priority Data: 197 34 413.5 8 August 1997 (08.08.97) DE</p> <p>(71) Applicant (for all designated States except US): TELEFON- AKTIEBOLAGET LM ERICSSON(publ) [SE/SE]; S-126 25 Stockholm (SE).</p> <p>(72) Inventor; and</p> <p>(75) Inventor/Applicant (for US only): SINANDER, Niklas [SE/SE]; Kantzegatan 22, S-421 47 Västra Frölunda (SE).</p> <p>(74) Agents: VON FISCHERN, Bernhard et al.; Hoffmann.Eitl, Arabellastrasse 4, D-81925 München (DE).</p>		<p>(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, US, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).</p> <p>Published With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</p>	

(54) Title: METHOD FOR UPGRADING A DATABASE



(57) Abstract

When upgrading a database the downtime of the database should be kept to a minimum as well as the hardware and software requirements. Moreover, a rollback of the upgrade should be possible at any time. The method disclosed here comprises the steps of creating a new table in said database, copying and/or transforming data from said table to said new table, adding a new version of said stored procedure to said database, and adding an additional stored procedure to the database which refers to both the previous version and the new version of the stored procedure whenever data is to be processed with the stored procedure, for causing data processing to take place in accordance with the previous version and the new version of the stored procedure.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NE	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NL	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NO	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NZ	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PL	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakhstan	RO	Romania		
CU	Cuba	LC	Saint Lucia	RU	Russian Federation		
CZ	Czech Republic	LI	Liechtenstein	SD	Sudan		
DE	Germany	LK	Sri Lanka	SE	Sweden		
DK	Denmark	LR	Liberia	SG	Singapore		
EE	Estonia						

METHOD FOR UPGRADING A DATABASE

5

The invention relates to a method for upgrading databases including a change of the database format.

10 Relational databases are used to handle the storage and retrieval of data for different applications and environments. Databases of this kind run on computer systems. When new functionality shall be added to a database, i.e when the database is upgraded, the format in
15 which the data is stored very often has to be changed as well. Usually, during the change access to the database must be denied to avoid inconsistency. In other words, it is not allowed to add or change (update) data while the change of the database functionality and format takes
20 place. At least, the database is not accessible during certain phases of an upgrade which phases should be minimized.

Further, if the upgrade of the database fails it must be
25 possible to return to the previous state of the database. Data changes that have been applied during or after the upgrade have to be taken into account as well as data changes queued to be applied.

30 A conventional solution to the above mentioned problem is to use an additional system during the upgrade. The additional system is a fully operable system with all the functionality of the original system. Extra hardware and software is required to log all data changes during the
35 upgrade of the database. Before the upgrade, the database is copied to the additional system, and a logging mechanism is initiated, for example, in the original system. The

upgrade of the database is then performed in the additional system while the original system continues to operate as before but also logs all data changes during the upgrade. When the database upgrade is finished, all logged data 5 changes are applied to the upgraded database in the additional system. If the upgrade fails, the additional system is simply disconnected and the original system continues with its normal operation, however, without additionally logging data changes any longer.

10

The above method for upgrading a database is disadvantageous since an additional system has to be installed and brought to the site where the database is to be upgraded. During the upgrade double software licenses 15 are need since on both the original and the additional system databases run in parallel for a certain time. Further, it turns out to be quite complex to manage the database upgrade due to all manual intervention necessary.

20 Therefore, the problem to be solved by the invention is to provide a method for upgrading a database including a change of the database format which method avoids extra hardware and software, limits the interruption of the database operation to a minimum, and allows a save return 25 to the original state of the database in case of a failure during the upgrade.

The above problem is solved by a method for upgrading a database using a table for storing data and a stored 30 procedure for processing data stored in said table of said database, said database running on a computer system, comprising the steps of:

- creating new a table in said database;
- copying and/or transforming data from said table to 35 said new table;
- adding a new version of said stored procedure to said database; and

- adding an additional stored procedure to the database which refers to both the previous version and the new version of the stored procedure whenever data is to be processed with the stored procedure, for causing data processing to take place in accordance with the previous version and the new version of the stored procedure.

5 For creating a link to said additional stored procedure a systemtable can be used. The systemtable can be a table of the database. As an alternative or in addition a configuration file can be used for creating a link to said additional stored procedure. The configuration file can be stored on a storage medium of the computer system.

10 15 The above problem is further solved by a method for upgrading a database by replacing an old version with a new version of a component of the database providing database functionality, said database running on a computer system wherein during the upgrade the old version and the new version of the database component providing database functionality are used at least temporarily in parallel and wherein reference to both versions is made through an additional component of the database providing a link to both versions of the database component.

20 25 The link can be provided by means of a systemtable referring an access to said component of the database to said additional component. The systemtable can be a table of the database.

30 35 In addition or as an alternative the link can be provided by means of a configuration file referring an access to said component of the database to said additional component. The configuration file is advantageously stored on a storage medium of said computer system.

The components of the database are stored procedures, trigger, indexes or tables. The new version of the table may have a format different from the format of the old version of the table.

5

The invention will be described in greater detail further below where reference is made to the drawings.

10 Figure 1 shows a diagramm of a computer system hosting a database and workstations accessing the database.

Figure 2a shows a diagramm illustrating an upgrade of a database on table level according to the invention.

15 Figure 2b shows a diagramm illustrating an upgrade of a database on page level according to the invention.

20 Figure 3 shows a diagramm of a computer system using a systemtable for upgrading a database according to the invention.

Figure 4 shows a diagramm of a computer system using a configuration file for upgrading a database according to the invention.

25

Figure 1 shows a computer system 1 hosting a database 2 on an appropriate storage medium, for example a hard disk or a group of hard disks. Users access the information or data stored in the database, for example, from a terminal or 30 from their workstations 3a, 3b, ... which are connected to the computer system 1 either directly or via a local or wide area network (LAN or WAN). The database 2 comprises tables, stored procedures, triggers, indexes etc. all of which provide certain database functions, respectively.

35

When an upgrade of the database is required, changes will be made to all or some of the components providing database functionality. Usually new and/or additional functions are provided by means of new and/or additional stored 5 procedures, triggers and so on. This sometimes also requires that changes are made to the format of the tables of the database.

According to the invention new tables are created to receive the data stored in the old tables, which are the tables used by the database until the upgrade of the database. Data are copied and/or transformed into the new tables during which process the data are adapted to the new format of the tables if required. To avoid downtime, i.e. 10 periods during which the users cannot access the database, the process of copying and/or transforming data from the old tables to the new tables needs special consideration. According to the invention, data sets of a predetermined 15 granularity are defined within the tables which data sets are handled jointly during the upgrade process to reduce the time periods during which the system is not accessible. 20

As shown in Figure 2a, if an upgrade is based on table level (granularity I), the whole table to be upgraded is modified in one operation. This approach should be chosen where the modification is a fast operation, for example adding a column with a default value. While a table is upgraded, access to the table as a whole is denied to avoid inconsistency. As shown in Figure 2b, if the modification 25 of a table is a more complicated operation that takes longer time, the table is upgraded on page level (granularity II). A page can be understood to be an internal system unit of a specific size, for example 1024 30 Bytes, which usually comprises - completely or partly - one or more related information units such as fields or records. By denying access to a page, one or more records 35 of a table are locked. For upgrading on page level, a new

table is installed and data from the old table is copied and/or transformed to the new table as indicated by the operator T in Figure 2b. While the table is upgraded, access to only the page currently processed by operator T 5 is denied to avoid inconsistency.

Moreover, both the old and the new tables have to be synchronized as far as data changes and new entries are concerned. By updating both the new and the old tables, as 10 proposed by the invention, not only data consistency is achieved but also the option to dismiss the upgrade and continue operation of the original system. The update of data in the old and the new table is an atomic operation, i.e. access is denied to the table or the page until both 15 tables are updated. Preferably, an update of data is performed in the new tables only where data sets have already been copied and/or transformed from the old tables to the new tables.

20 To perform an upgrade of tables of a database according to the invention, the remaining components providing database functionality like stored procedures, trigger, indexes, etc. have to exist in an old version as well as in a new version. To avoid any conflict while accessing a component 25 of the database, like stored procedures, a method for handling different versions of stored procedures or the like is required. Three methods will be discussed in the following.

30 A first method for handling versions of database components will be described with special emphasis on stored procedures of the database. The skilled person will understand that this process can also be performed for other components of a database comparable to stored 35 procedures.

As shown in Figure 3 a systemtable is used during the upgrade process which systemtable holds references to all or just thoses stored procedure which are to be upgraded, i.e. replaced by stored procedure of a new version. All or 5 just thoses stored procedure which are to be upgraded are referenced through the systemtable during the upgrade process. A systemtable according to the invention may look as follows:

10

Table 1

	name of stored procedure known to the system	base version	target version	upgrade version
15	sp_a	sp_a_1.0	sp_a_1.1	sp_a_upgr
	sp_b	sp_b	-	-
20	sp_c	sp_c_1.0	sp_c_1.1	sp_c_upgr

During normal operation the entries in the systemtable refers only to one set of stored procedures, i.e. the base 25 version. When a workstation accesses the database the system looks up the momentarily valid name or version of the stored procedure to which the specific transaction initiated by the workstation is linked. For example, if the access from a workstation requires that stored procedure 30 sp_a is processed the system will look up the valid version of stored procedure sp_a in the systemtable. In the above table and during normal operation, the valid version of stored procedure sp_a is sp_a_1.0 which is then used by the system to perform the transaction requested by the 35 workstation.

In case of an upgrade of the database, new versions of all or some stored procedures are to be installed in the database. It is also possible that a new stored procedure is added to the database. According to the invention, the 5 systemtable is updated with names/versions of the new stored procedures, i.e. the target versions are added. To keep old and new tables of the database synchronized additional stored procedures `sp_a_upgr` and `sp_c_upgr` are added to the systemtable. The additional stored procedures 10 `sp_a_upgr` and `sp_c_upgr` are used only during the upgrade process and use the new and the old stored procedures to update both the new and the old tables, respectively, in accordance with transactions initiated by the workstations.

15 For example, during an upgrade the additional stored procedure `sp_a_upgr` initiates in the database two branches of processing on the basis of base version `sp_a_1.0` and new version `sp_a_1.1` whereby data is processed and updated in both the new and the old tables of the database system.

20 With respect to stored procedure `sp_b`, there exists no additional stored procedure since stored procedure `sp_b` is not upgraded, i.e. remains the same in the old and the new database. The system will refer to the basic version `sp_b` in such case and use the base version for both the old and 25 the new tables.

30 The switch over from the old stored procedures to the new stored procedures would be to deny access to the systemtable, update the references so that the desired stored procedures are referenced and to grant access to the systemtable. The updating of references is a very fast operation, so the users of the system will only experience 35 a small delay and no downtime.

A second method for handling versions of database components will be described in the following. As shown in Figure 4 a configuration file is used during the upgrade process which file holds references to all or just those stored procedure which are to be upgraded, i.e. replace by stored procedure of a new version. All or just those stored procedure which are to be upgraded are referenced through the configuration file during the upgrade process. The configuration file is stored on a separate storage media 4 provided in the computer system 1 or the storage media on which the database 2 is also stored.

Before the database is accessed by an entity (e.g. an application) from a workstation 3, the configuration file is read to see which stored procedure to use. If there is no entry in the configuration file for a specific stored procedure, the stored procedure as known to the system is used. A configuration file according to the invention can be a simple text file and may look as follows:

20

Table 2

```
# UPGRADE CONFIGURATION FILE
#
25      #ENTRIES: name of stored procedure known to the
          system; #base version; target version; upgrade
          version
          sp_a; sp_a_1.0; sp_a_1.1; sp_a_upgr
          sp_b; sp_b
30      sp_c; sp_c_1.0; sp_c_1.1; sp_c_upgr
          ...
During normal operation the entries in the configuration
file refer only to one set of stored procedures, i.e. the
base version. When a workstation accesses the database the
35    system looks up the momentarily valid name or version of
          the stored procedure to which the specific transaction
          initiated by the workstation is linked. For example, if the
```

access from a workstation requires that stored procedure `sp_a` is processed the system will look up the valid version of stored procedure `sp_a` in the configuration file. In the above file, the valid version of stored procedure `sp_a` is 5 `sp_a_1.0` which is then used by the system to perform the transaction requested by the workstation.

In case of an upgrade of the database, new versions of all or some stored procedures are to be installed in the 10 database. It is also possible that a new stored procedure is added to the database. According to the invention, the configuration file is updated with names/versions of the new stored procedures, i.e. the target versions are added. To keep old and new tables of the database synchronized 15 additional stored procedures `sp_a_upgr` and `sp_c_upgr` are added to the configuration file. The additional stored procedures `sp_a_upgr` and `sp_c_upgr` are used only during the upgrade process and use the new and the old stored procedures to update both the new and the old tables, 20 respectively, in accordance with transactions initiated by the workstations.

For example, during an upgrade the additional stored procedure `sp_a_upgr` initiates at the workstation two branches 25 of processing on the basis of base version `sp_a_1.0` and new version `sp_a_1.1` whereby data is processed and updated in both the new and the old tables of the database system. With respect to stored procedure `sp_b`, there exists no 30 additional stored procedure since stored procedure `sp_b` is not upgraded, i.e. remains the same in the old and the new database. The system will refer to the basic version `sp_b` in such case and use the base version for both the old and the new tables.

configuration file and reread it. To update and reread the configuration file is a very fast operation, so that the users of the system will only experience a small delay and no downtime.

5

A third method for handling versions of database components could be a combination of a systemtable and a configuration file. The systemtable will refer to some stored procedures while the configuration file will refer to other stored

10 procedures.

The other components of the database can be handled in the same manner. For example, to be able to hold two or more versions of tables, the tables could be referenced through

15 the systemtable or configuration file and treated similarly as outlined above with respect of stored procedures.

Claims

1. Method for upgrading a database using a table for
5 storing data and a stored procedure for processing data
stored in said table of said database, said database
running on a computer system, comprising the steps of:

- creating new a table in said database;
- 10 - copying and/or transforming data from said table to
said new table;
- adding a new version (sp_a_1.1, sp_b_1.1) of said
stored procedure to said database; and
- adding an additional stored procedure (sp_a_upgr,
15 sp_b_upgr, ...) to the database which refers to both
the previous version (sp_a_1.0, sp_b_1.0) and the new
version (sp_a_1.1, sp_b_1.1) of the stored procedure
whenever data is to be processed with the stored
procedure, for causing data processing to take place
20 in accordance with the previous version (sp_a_1.0,
sp_b_1.0) and the new version (sp_a_1.1, sp_b_1.1) of
the stored procedure.

2. Method for upgrading a database according to claim 1
25 characterized in that a systemtable creates a link to said
additional stored procedure (sp_a_upgr, sp_c_upgr).

3. Method for upgrading a database according to claim 2
characterized in that said systemtable is a table of said
30 database.

4. Method for upgrading a database according to claim 1,
2 or 3 characterized in that a configuration file creates a
link to said additional stored procedure (sp_a_upgr,
35 sp_c_upgr).

5. Method for upgrading a database according to claim 4 characterized in that said configuration file is stored on a storage medium (4) of said computer system.

5 6. Method for upgrading a database by replacing an old version with a new version of a component of the database providing database functionality, said database running on a computer system characterized in that during the upgrade the old version (sp_a_1.0, sp_c_1.0) and the new version 10 (sp_a_1.1, sp_c_1.1) of the database component providing database functionality are used at least temporarily in parallel and that reference to both versions is made through an additional component (sp_a_upgr, sp_b_upgr) of the database providing a link to both versions of the 15 database component.

7. Method according to claim 6 characterized in that said link is provided by means of a systemtable referring an access to said component (sp_a, sp_c) of the database to 20 said additional component (sp_a_upgr, sp_c_upgr).

8. Method according to claim 7 characterized in that said systemtable is a table of said database.

25 9. Method according to claim 6, 7 or 8 characterized in that said link is provided by means of a configuration file referring an access to said component (sp_a, sp_c) of the database to said additional component (sp_a_upgr, sp_c_upgr).

30 10. Method according to claim 9 characterized in that said configuration file is stored on a storage medium (4) of said computer system.

35 11. Method according to any one of claims 6 to 10 characterized in said component of the database is a stored procedure.

12. Method according to any one of claims 6 to 10 characterized in that said component of the database is a trigger.

5

13. Method according to any one of claims 6 to 10 characterized in that said component of the database is an index.

10

14. Method according to any one of claims 6 to 10 characterized in that said component of the database is a table.

15

15. Method according to claim 14 characterized in that the new version of said table has a format different from the format of the old version of said table.

20

16. Method for upgrading a database using a table for storing data, said database running on a computer system, comprising the steps of:

25

- creating a new table in said database;
- defining data sets of a predetermined granularity within the table on the basis of the time required to copy and/or transform data to the new table, during which time the data set can not be accessed; and
- copying and/or transforming data from said table to said new table, wherein data sets are handled jointly.

30

17. Method according to claim 16 characterized in that said data sets are defined to be the table.

18. Method according to claim 16 characterized in that said data sets are defined to be a page of the table.

35

19. Method according any one of claims 16 to 18 characterized in that said database using a stored

procedure for processing data stored in said table of said database, comprising the steps of:

- adding a new version (sp_a_1.1, sp_b_1.1) of said stored procedure to said database; and
- adding an additional stored procedure (sp_a_upgr, sp_b_upgr, ...) to the database which refers to both the previous version (sp_a_1.0, sp_b_1.0) and the new version (sp_a_1.1, sp_b_1.1) of the stored procedure whenever data is to be processed with the stored procedure, for causing data processing to take place in accordance with the previous version (sp_a_1.0, sp_b_1.0) and the new version (sp_a_1.1, sp_b_1.1) of the stored procedure.

15

20. Method for upgrading a database according to claim 19 characterized in that a systemtable creates a link to said additional stored procedure (sp_a_upgr, sp_c_upgr).

20 21. Method for upgrading a database according to claim 20 characterized in that said systemtable is a table of said database.

25 22. Method for upgrading a database according to claim 19, 20 or 21 characterized in that a configuration file creates a link to said additional stored procedure (sp_a_upgr, sp_c_upgr).

23. Method for upgrading a database according to claim 22 characterized in that said configuration file is stored on a storage medium (4) of said computer system.

30 24. Method for upgrading a database according to claims 16 to 18, comprising the step of replacing an old version with a new version of a component of the database providing database functionality, wherein during the upgrade the old version (sp_a_1.0, sp_c_1.0) and the new version (sp_a_1.1,

sp_c_1.1) of the database component providing database functionality are used at least temporarily in parallel and that reference to both versions is made through an additional component (sp_a_upgr, sp_b_upgr) of the database 5 providing a link to both versions of the database component.

25. Method according to claim 24 characterized in that said link is provided by means of a systemtable referring 10 an access to said component (sp_a, sp_c) of the database to said additional component (sp_a_upgr, sp_c_upgr).

26. Method according to claim 25 characterized in that said systemtable is a table of said database.

15 27. Method according to claim 24, 25 or 26 characterized in that said link is provided by means of a configuration file referring an access to said component (sp_a, sp_c) of the database to said additional component (sp_a_upgr, 20 sp_c_upgr).

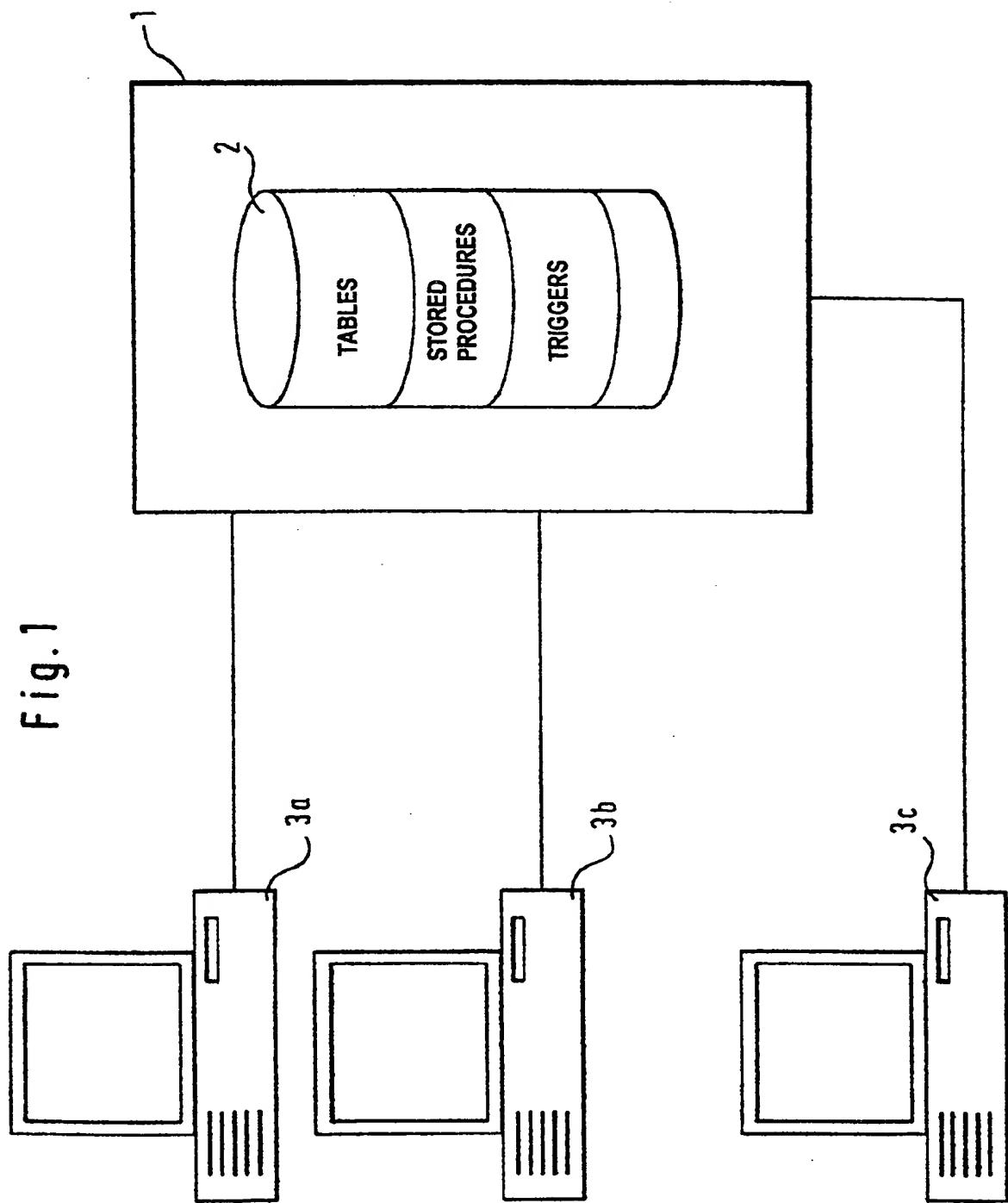
28. Method according to claim 27 characterized in that said configuration file is stored on a storage medium (4) of said computer system.

25 29. Method according to any one of claims 24 to 28 characterized in said component of the database is a stored procedure.

30 30. Method according to any one of claims 24 to 28 characterized in that said component of the database is a trigger.

35 31. Method according to any one of claims 24 to 28 characterized in that said component of the database is an index.

32. Method according to any one of claims 6 to 28 characterized in that said component of the database is a table.



2/4

Fig. 2a

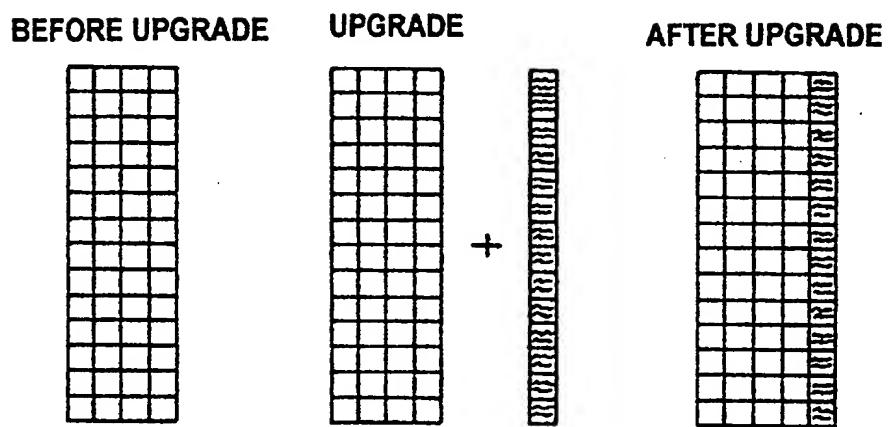


Fig. 2b

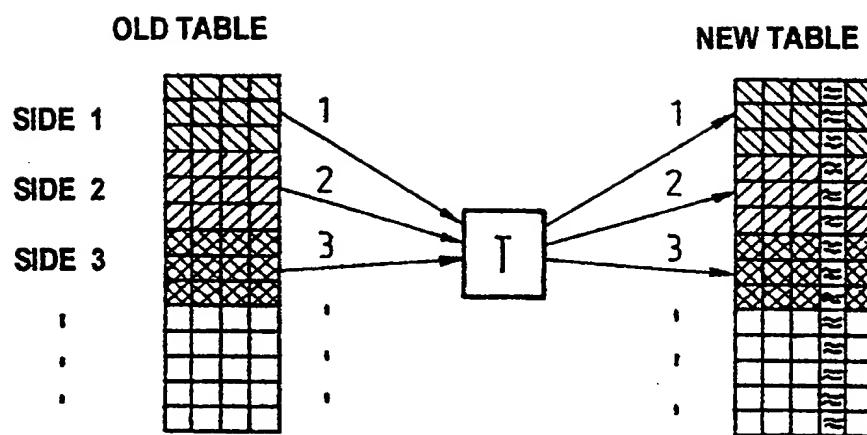
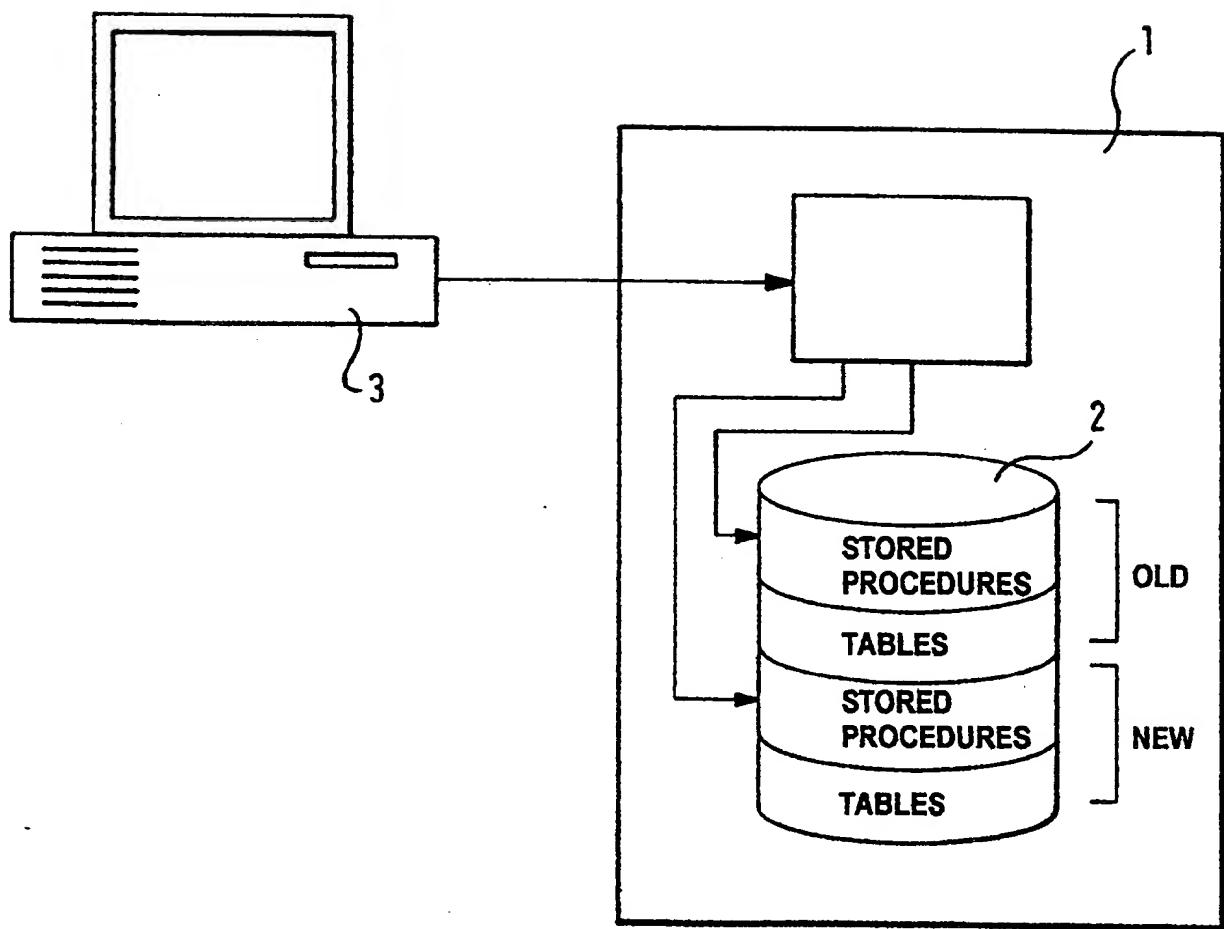
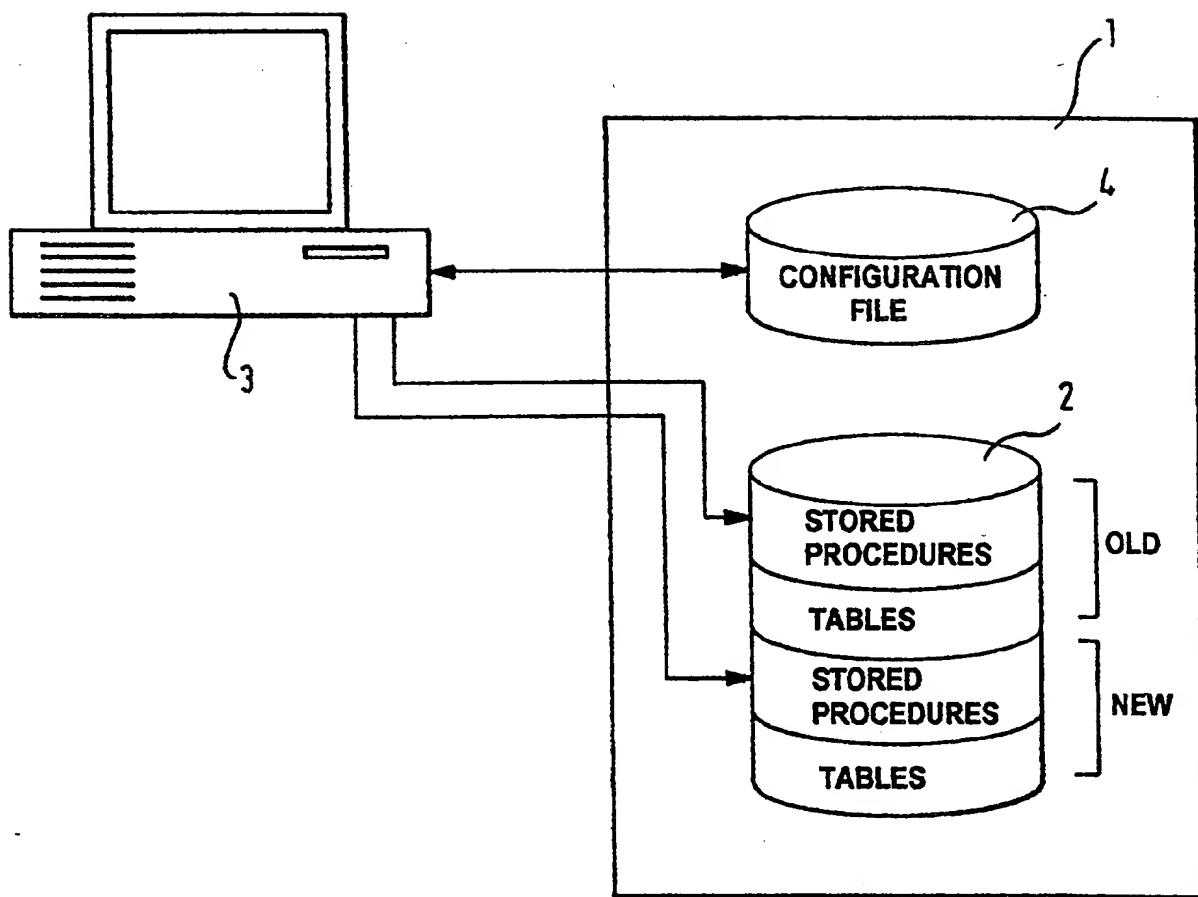


Fig. 3



4/4

Fig.4



INTERNATIONAL SEARCH REPORT

International Application No.

PCT/EP 98/05026

A. CLASSIFICATION OF SUBJECT MATTER

IPC 6 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 396 623 A (MCCALL ANNETTE B ET AL) 7 March 1995 see abstract see column 3, line 66 - column 4, line 16 see column 13, line 26 - column 14, line 61 see figures 3,4 ---	1-3,6,16
A	WO 97 14091 A (IBM) 17 April 1997 see abstract see claims ---	1,6,16
A	US 5 359 730 A (MARRON ASSAF) 25 October 1994 see abstract see column 2, line 52 - column 5, line 53 ----	1,6,16



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

° Special categories of cited documents :

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

Date of mailing of the international search report

5 January 1999

13/01/1999

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl.
Fax: (+31-70) 340-3016

Authorized officer

Abbing, R

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No-

PCT/EP 98/05026

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US 5396623	A 07-03-1995	NONE		
WO 9714091	A 17-04-1997	EP 0855056 A 29-07-1998	JP 10512079 T 17-11-1998	
US 5359730	A 25-10-1994	NONE		